

Amendments to the claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A system for evaluating tests of a computer program, comprising:
  - one or more processors;
  - a computer-readable storage medium containing instructions configured to cause the one or more processors to perform operations, including:
    - creating a file containing a list of required actions for testing an application, wherein the file includes automation control code, and wherein the file is created by a test designer on a test design automation independent tool within a first insulated environment;
    - automatically transmitting the file to a test coding and execution automator in a second insulated environment that is separate from the first insulated environment;
    - receiving the file in the second insulated environment, wherein an interface is generated through which the test coding and execution automator enters one or more code segments, and uses an automation engine to generate test automation code by associating the one or more code segments with one or more details abstracted from the automation control code;
    - using an automation framework to translate the test automation code into an automation application map by mapping the test automation code to the automation application map, wherein the automation application map includes a lower level of executable computer code used to test the application;
    - executing the automation application map code ~~test automation code~~ to test the application and produce test results, ~~wherein the test automation code is executed using the automation application map;~~
    - automatically transmitting the test results to a test analysis automation independent tool in a third insulated environment that is separate from the first and second insulated environments; and

receiving the test results in the third insulated environment and comparing the received test results to one or more other received test results, wherein the comparison is done by a test results analyst using the test analysis automation independent tool in the third insulated environment.

2 - 6. (Cancelled).

7. (Previously Presented) The system of claim 1, wherein the automatically transmitting of the test results to the test analysis automation independent tool in the third insulated environment involves copying or moving the test results from the second insulated environment to the third insulated environment.

8. (Previously Presented) The system of claim 7, wherein the third insulated environment includes an internet web browser for viewing received test results.

9. (Previously Presented) The system of claim 1, wherein the format of the received test results include JPEG, HTML, GIF, and combinations thereof.

10 - 15. (Cancelled).

16. (Previously Presented) The system of claim 1, wherein the list includes an action to test a computer-human interface generated by the computer program.

17. (Previously Presented) The system of claim 1, wherein the list includes an action to test performance of the computer program.

18 - 24. (Cancelled).

25. (Previously Presented) The system of claim 1, wherein the first insulated environment, the second insulated environment, and the third insulated environment operate on a network such that each of the environments are accessible through different

computer terminals, wherein the first insulated environment operates on a first computer which does not contain the second insulated environment or the third insulated environment, wherein the second insulated environment operates on a second computer which does not contain the first insulated environment or the third insulated environment, and wherein the third insulated environment operates on a third computer which does not contain the first insulated environment or the second insulated environment.

26. (Currently Amended) A computer-implemented method for evaluating tests of a computer program, comprising:

creating a file containing a list of required actions for testing an application, wherein the file includes automation control code, and wherein the file is created by a test designer on a test design automation independent tool within a first insulated environment;

automatically transmitting the file to a test coding and execution automator in a second insulated environment that is separate from the first insulated environment;

receiving the file in the second insulated environment, wherein an interface is generated through which the test coding and execution automator enters one or more code segments, and uses an automation engine to generate test automation code by associating the one or more code segments with one or more details abstracted from the automation control code;

using an automation framework to translate the test automation code into an automation application map by mapping the test automation code to the automation application map, wherein the automation application map includes a lower level of executable computer code used to test the application;

executing the automation application map code ~~test automation code~~ to test the application and produce test results, ~~wherein the test automation code is executed using the automation application map;~~

automatically transmitting the test results to a test analysis automation independent tool in a third insulated environment that is separate from the first and second insulated environments; and

receiving the test results in the third insulated environment and comparing the received test results to one or more other received test results, wherein the comparison is done by a test results analyst using the test analysis automation independent tool in the third insulated environment.

27. (Currently Amended) Computer software stored on one or more computer-readable storage mediums, the computer software comprising program code for carrying out a method for evaluating tests of a computer program, the method comprising:

creating a file containing a list of required actions for testing an application, wherein the file includes automation control code, and wherein the file is created by a test designer on a test design automation independent tool within a first insulated environment;

automatically transmitting the file to a test coding and execution automator in a second insulated environment that is separate from the first insulated environment;

receiving the file in the second insulated environment, wherein an interface is generated through which the test coding and execution automator enters one or more code segments, and uses an automation engine to generate test automation code by associating the one or more code segments with one or more details abstracted from the automation control code;

using an automation framework to translate the test automation code into an automation application map by mapping the test automation code to the automation application map, wherein the automation application map includes a lower level of executable computer code used to test the application;

executing the automation application map code ~~test automation code~~ to test the application and produce test results, ~~wherein the test automation code is executed using the automation application map;~~

automatically transmitting the test results to a test analysis automation independent tool in a third insulated environment that is separate from the first and second insulated environments; and

receiving the test results in the third insulated environment and comparing the received test results to one or more other received test results, wherein the comparison is

done by a test results analyst using the test analysis automation independent tool in the third insulated environment.

28 - 29. (Cancelled).